PATENT
IBM Docket No. CA9-2000-0042US1

Listing of the Claims (including amendments and status):

1   1.(Currently amended) A method for testing a software test
2   component, said method comprising the steps of:

3   ascertaining a public interface of the software test
4   component; and

5   creating a wrapper component for the software test component
6   by the substeps of

7   defining a wrapper component interface to mirror the
8   public interface of the software test component and to
9   receive calls to the software test component, defining
10   the wrapper component to delegate to the software test
11   component by including calls to the public interface of
12   the software test component within the wrapper
13   component, inserting test code within the wrapper
14   component to permit capture and playback of user
15   interaction with the public interface of the software
16   test component, and

17   enabling a test case to use the wrapper component
18   interface to ~~access~~ pass the received calls to the
19   software test component and to generate test data from
20   the test code in the wrapper component.

1   2.   (Previously Presented) The method for testing a software
2   test component according to Claim 1, wherein the software test
3   component is an object-oriented software test component and

- 2 -

4 wherein said step of ascertaining the public interface of the
5 software test component further comprises interrogating a test
6 component definition to determine public methods, constructor and
7 associated parameters for the software test component.

1 3. (Previously Presented) The method for testing a software
2 test component according to Claim 2, wherein the test component
3 is a Java language class and said ascertaining step further
4 comprises use of an introspection group of interfaces in a Java
5 Bean specification.

1 4. (Previously presented) The method for testing a software test
2 component according to Claim 2, wherein the substep of defining a
3 wrapper component interface further comprises defining public
4 methods, constructors and associated parameters in the wrapper
5 component to mirror the public methods, constructors and
6 parameters determined for the software test component.

1 5. (Currently amended) A computer <u>memory storing logic</u> ~~program~~
2 ~~product~~ for testing software components, the <u>logic</u> ~~computer~~
3 ~~program product comprising a computer usable medium having~~
4 ~~computer readable code means embodied in said medium,~~ comprising:

5   first <u>logic</u> ~~code means~~ for ascertaining a public interface
6   of a software test component; and

-3-

8   second <u>logic</u> ~~code means~~ for creating a wrapper component for

9   the software test component, said second code means

10   comprising

11

12   <u>logic</u> ~~code means~~ for defining a wrapper component

13   interface to mirror the public interface of the

14   software test component,

15   <u>logic</u> ~~code means~~ for defining the wrapper component to

16   delegate to the software test component by including

17   calls to the public interface of the software test

18   component within the wrapper component,

19   <u>logic</u> ~~code means~~ for inserting test code within the

20   wrapper component to permit capture and playback of

21   user interaction with the public interface of the

22   software test component, and

23   <u>logic</u> ~~code means~~ for enabling a test case to use the

24   wrapper component interface to access the software test

25   component and <u>then</u> to generate test data from the test

26   code in the wrapper component.

1   6. (Currently amended) A system in a computing environment for

2   generation of a test environment for a software test component,

3   said system comprising:

4

5   means for ascertaining a public interface of the software

6   test component; and

-4-

7   means for creating a wrapper component for the software test
8   component, said creating means including

9   means for defining a wrapper component interface to
10   mirror the public interface of the software test
11   component,

12   means for defining the wrapper component to delegate to
13   the software test component by including calls to the
14   public interface of the software test component within
15   the wrapper component, and

16   means for making the wrapper component available to
17   permit test code to be inserted within the wrapper
18   component to permit capture and playback of user
19   interaction with the public interface of the software
20   test component and to permit a test case component to
21   use the wrapper component interface to ~~access~~ <u>pass data</u>
22   <u>to</u> the software test component and to generate test
23   data from the test code in the wrapper component.

1   7.   (Previously Presented) The system for generation of a test
2   environment according to Claim 6, wherein the software test
3   component is an object-oriented software test component and said

4   means for ascertaining the public interface of the software test
5   component further comprises

-5

6   means for interrogating a definition of the software test

7   component to determine public methods,

8   constructor and associated parameters for the software test

9   component.

1   8.   (Previously Presented) The system for generation of a test

2   environment according to Claim 7, wherein the means for defining

3   a wrapper component interface comprises means for defining

4   public methods, constructors and associated parameters in the

5   wrapper component to minor the

6   public methods, constructors and parameters ascertained for the

7   software test component.

- 6 -